UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/689,417	10/20/2003	Manoj K. Agarwal	JP920030099US1	2532

7590 12/13/2007

Frederick W. Gibb, III
McGinn & Gibb, PLLC
Suite 304
2568-A Riva Road
Annapolis, MD 21401

EXAMINER	
KANG, INSUN	

ART UNIT	PAPER NUMBER
2193	

MAIL DATE	DELIVERY MODE
12/13/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/689,417	AGARWAL ET AL.	
	Examiner	Art Unit	
	Insun Kang	2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *23 November 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-4 and 6-35* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-4, 6-10, and 16-35* is/are rejected.

7)☒ Claim(s) *11-15* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This action is in response to the amendment filed on 11/23/2007.

2. As per applicant's request, claims 1-4 and 6-35 have been amended. Claims 1-4 and 6-35 are pending in the application.

Allowable Subject Matter

3. Claims 11-15 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1-4, 6-10, 20-25, and 30-35 are rejected under 35 U.S.C. 103(a) as being unpatentable over Keller et al. ("An Active Approach to Characterizing Dynamic Dependencies for Problem Determination in a Distributed Environment," 2001) hereafter Keller in view of The Open Group ("Systems Management: Distributed Software Administration," published on 1/1998).

Per claim1:

Keller discloses the invention as claimed including a method for determining run-time dependencies between logical components (i.e. sec 4.1, p5, lines 7-8),

- monitoring run-time activity of each of a first logical component and a second logical component of the data processing environment, said monitoring comprising determining a first activity period for said first and a second activity period for said second logical component (i.e. section 4.2 Detecting and Characterizing Operational Dependencies, page 5, lines 4-9; i.e. page 6, step 4, lines 1-15, , page 7 Fig 2)

- comparing the monitored run-time activity of the first logical component with the monitored run-time activity of the second logical component to identify correlations between the monitored run-time activity of the first and second logical components (i.e. page 6, step 4, lines 1-13, page 7 Fig 2);

- said comparing comprising comparing said first activity period and said second activity period to identify correlations between said first logical component and said second logical component (i.e. "a regression line relating the mean **log response time** to the perturbation level. The regression line is the key to automatically identifying and characterizing dependencies, page 10 sec. 5.5. Data analysis, last two lines; page 5 4.2 Active Dependency Discovery steps 1-4).

- and in response to identification of a positive correlation between the monitored run-time activity of the first and second logical components, recording the existence of a dependency relationship between the first and second logical components (i.e. page 6, step 4, lines 1-15, , page 7 Fig 2; page 5, 4.2 Active Dependency Discovery steps 1-4; "collecting as much of the dependency model as possible," page 5, lines 12-13; see Figure 2 in page 7; "each such graph

maps the dependencies between one user interaction and the particular database tables upon which that interaction depends," page 2, third paragraph).

Keller does not explicitly teach that determining comprises determining a first start time and a first end time for said first logical component and a second start time and a second end time for said second logical component and comparing comprises determining whether said first start time is before said second start time and whether said first end time is after said second end time. However, The Open Group teaches that determining software object containment was known in the pertinent art, at the time applicant's invention was made, to "designate objects that are defined within the context of their containing objects (i.e. page 172, last paragraph)." It would have been obvious for one having ordinary skill in the art to modify Keller's disclosed system to incorporate the teachings of The Open Group. The modification would be obvious because one having ordinary skill in the art would be motivated to identify the component correlations by the containment association among the components.

Per claim 2:

Keller further teaches:

- monitoring run-time activity data of each of a plurality of logical components additional to said first and second logical components(i.e. section 4.2 Detecting and Characterizing Operational Dependencies, page 5, lines 4-9) ;

- comparing the monitored run-time activity of the first logical component with the monitored run-time activity of the plurality of logical components to identify positive correlations between the monitored run-time activity of the first logical component and the monitored run-time

activity of any of the plurality of logical components; (i.e. page 6, step 4, lines 1-13, page 7 Fig 2);

-and recording the existence of a dependency relationship between the first logical component and any of the plurality of logical components for which a positive correlation is identified (i.e. page 6, step, lines 1-13, page 7 Fig 2; page 5, section 4.2 Active Dependency Discovery steps 1-4; "collecting as much of the dependency model as possible," page 5, lines 12-13; see Figure 2 in page 7; "each such graph maps the dependencies between one user interaction and the particular database tables upon which that interaction depends," page 2, third paragraph).

Per claim 3:

Keller further teaches:

- aggregating the recorded dependency relationships of the first logical component(i.e. "collecting as much of the dependency model as possible," page 5, lines 12-13; see Figure 2 in page 7; "each such graph maps the dependencies between one user interaction and the particular database tables upon which that interaction depends," page 2, third paragraph).

Per claim 4:

Keller further teaches:

-comparing the monitored run-time activity of the second logical component with the run-time activity of each of the plurality of logical components to identify positive correlations between the monitored run-time activity of the second logical component and the monitored run-time

activity of any of the plurality of logical components (i.e. page 6, step 4, lines 1-13, page 7 Fig 2);

- recording the existence of a dependency relationship between the second logical component and any of the plurality of logical components for which a positive correlation is identified (i.e. page 6, step 4, lines 1-13, page 7 Fig 2; page 5, section 4.2 Active Dependency Discovery steps 1-4; "collecting as much of the dependency model as possible," page 5, lines 12-13; see Figure 2 in page 7; "each such graph maps the dependencies between one user interaction and the particular database tables upon which that interaction depends," page 2, third paragraph).

Per claim 6:

Keller further teaches:

- wherein the comparing step comprises determining whether the activity period of the first logical component contains the activity period of the second logical component (i.e. "where I indexes the transaction type...is the value of the **mean log response time for the k'th execution of the i'th transaction type**, j indexes the 10 database table," page 11, lines 9-11; "a regression line relating the mean **log response time** to the perturbation level. The regression line is the key to automatically identifying and characterizing dependencies, page 10 sec. 5.5. Data analysis, last two lines; page 5 4.2 Active Dependency Discovery steps 1-4).

Per claim 7:

Keller further teaches:

- wherein the step of monitoring run-time activity comprises monitoring invocations of the first and second logical components, and the comparing step comprises comparing the number of invocations of each of the first and second logical components within a monitoring period (i.e. "if αij is statistically non-zero..., then we conclude that transaction type i depends on table j with a strength of αij, page 11 lines 15-17; "where **i indexes the transaction type (1....,14)...is the value of the mean log response time for the k'th execution of the i'th transaction type**, j indexes the 10 database table," page 11, lines 9-11; page 5 4.2 Active Dependency Discovery steps 1-4).

Per claims 8 and 9:

Keller further teaches:

-monitoring run-time activity for a plurality of executions of each of a first and a second logical component, comparing the monitored run-time activity for the plurality of executions of the second logical component with the monitored run-time activity for the plurality of executions of the first logical component (i.e. section 4.2 Detecting and Characterizing Operational Dependencies, page 5, lines 4-9; page 11, lines 9-11) ;

-determining a proportion of executions of the first and second logical component for which a positive correlation is identified between the compared run-time activity of the first and second logical components (i.e. "**With the data thus reduced and linearized**, it becomes easy to fit a regression line relating the mean log response time to the perturbation level...key to automatically identifying and characterizing dependencies: **a statistically nonzero slope for**

this line indicates the existence of a dependency, and the magnitude of the slope characterizes its strength," page 10, last three lines and page 11 first three lines);

-and recording in association with the recorded dependency relationship a value representing the determined proportion of executions of the first logical component for which a positive correlation is identified (i.e. page 6 step 4, lines 1-15, page 7 Fig 2; page 5 4.2 Active Dependency Discovery steps 1-4; "collecting as much of the dependency model as possible," page 5, lines 12-13; see Figure 2 in page 7; "each such graph maps the dependencies between one user interaction and the particular database tables upon which that interaction depends," page 2, third paragraph).

Per claim 10:

Keller further teaches:

-determining a proportion of executions of the second logical component for which a positive correlation is identified between the compared run-time activity of the first and second logical components (i.e. "**With the data thus reduced and linearized,** it becomes easy to fit a regression line relating the mean log response time to the perturbation level...key to automatically identifying and characterizing dependencies: **a statistically nonzero slope for this line indicates the existence of a dependency, and the magnitude of the slope characterizes its strength,**" page 10, last three lines and page 11 first three lines);

-and recording in association with the recorded dependency relationship a value representing the determined proportion of executions of the second logical component for which a positive correlation is identified (i.e. page 6 step 4, lines 1-15, page 7 Fig 2; page 5 4.2 Active

Dependency Discovery steps 1-4; "collecting as much of the dependency model as possible," page 5, lines 12-13; see Figure 2 in page 7; "each such graph maps the dependencies between one user interaction and the particular database tables upon which that interaction depends," page 2, third paragraph)

-and generating a weight value (i.e. "the dependency edge represented by **label w3** in Figure 1 has been expanded to include the operational dependency edges," page 7, third paragraph, lines 2-3, "The overall **dependency strength w3 represents a weighted average** of the strengths of the operational dependency edges..., with **the weights determined by** the typical applied workload," see the note recited in page 7; page 3, sec 2. Dependency Models, lines 3-4)

- comprising a combination function of the determined proportions of executions of the first and second logical components, and storing the weight value in association with the recorded dependency relationship(i.e. page 6, step 4, lines 1-15, page 7 Fig 2; page 5 4.2 Active Dependency Discovery steps 1-4; page 5, lines 12-13; see Figure 2 in page 7).

Per claim 20, this claim is another version of the claimed method discussed in claim 3, wherein all claim limitations also have been addressed and/or covered in cited areas as set forth the above.

Per claim 21:

Keller further teaches:

- sorting the dependencies into an order determined by a sorting heuristic, and the step of analyzing the set of logical components comprises analyzing components of the set of components in said determined order (i.e. the dependency weights sorted as {w1, w2, w3...w8} in Figure 1 in page 3 and the dependencies are sorted by their strength in Item table in Figure 5).

Per claim 22, this claim is another version of the claimed method discussed in claim 1, wherein all claim limitations also have been addressed and/or covered in cited areas as set forth the above.

Per claims 23 and 24, they are the device versions of claim 1, respectively, and are rejected for the same reasons set forth in connection with the rejection of claim 1 above.

Per claim 25, it is the apparatus version of claim 1, respectively, and is rejected for the same reasons set forth in connection with the rejection of claim 1 above.

Per claim 30-35, they are another versions of the claimed methods discussed in claims 1-4 and 6-9, wherein all claim limitations also have been addressed and/or covered in the cited areas as set forth the above.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claims 16-19 and 26-29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Keller et al. ("An Active Approach to Characterizing Dynamic Dependencies for Problem Determination in a Distributed Environment," 2001) hereafter Keller, in view of The Open Group ("Systems Management: Distributed Software Administration," published on 1/1998), and further in view of Kar et al. ("Dynamic Dependencies in Application Service Management," June 2000) hereafter Kar.

Per claim 16:

Keller disclose, "monitors for performance, availability, and any other relevant metrics (page 5, Step 2, line 1)" and active dependency discovery system (section 4.2, page 5). Keller and The Open Group do not explicitly teach that the disclosed system is implemented as an agent. However, Kar teaches generating an event in response to a monitoring agent polling the first and second components for information relating to the processing of requests and accessing run-time activity data via the monitoring interface was known in the pertinent art, at the time applicant's invention was made, to autonomously accomplish tasks on behalf of its user (i.e. "The application service agent is the focal point for managing an individual service offering...The agent receives event notifications from the MLMs through the platform API and updates the view of the service that it maintains," see page 6, section 5 Dependency Architecture, Figure 3 and lines 13-23; "a service agent obtains the configuration file generated at the application dependency analysis phase," page 6, right col. Last two lines; page 7, first two lines). It would

have been obvious for one having ordinary skill in the art to modify Keller and The Open Group's disclosed system to incorporate the teachings of Kar. The modification would be obvious because one having ordinary skill in the art would be motivated to obtain abilities such as autonomy, flexibility etc by using an agent concept that is applied in Kar (i.e. page 6, right col. Last two lines).

Per claim 17:

Kar further teaches:

- wherein the monitoring agent comprises an aggregator for aggregating run-time activity data accessed via the monitoring interface(i.e. "The functional dependencies yield a generic service model while the structural part provides detailed information on the involved component. While the functional model is stored at the MLMs and the service management agent maintains a structural view," section 5, right column, last paragraph).

Per claim 18:

Kar further teaches:

- wherein the monitoring agent comprises program code for computing run-time activity metrics from the run-time activity data accessed via the monitoring interface (i.e. "through the platform API, page 6, section 5, paragraph 2).

Per claim 19:

Kar further teaches:

- wherein the monitoring agent is configured to poll the first and second logical components via the monitoring interface (i.e. section 5, page 7 lines 32-42; page 6, section 5, paragraph 2).

Per claim 26:

Keller and The Open Group do not explicitly teach that the dependency generator is configured to generate a CIM_dependency class instance to represent an identified positive correlation, the apparatus further comprising a CIM object manager configured to record dependency information within a CIM repository in the data storage unit. However, Kar teaches that such a CIM data model was known in the pertinent art, at the time applicant's invention was made, to describe common definitions of managed elements or information across the enterprise (i.e. "In the Common Information Model (CIM), dependencies...being usually perceived as a specific kind of association...are modeled as classes," page 2, section 2.1, second paragraph, lines 1-6). It would have been obvious for one having ordinary skill in the art to modify Keller and The Open Group's disclosed system to incorporate the teachings of Kar. The modification would be obvious because one having ordinary skill in the art would be to define the associations between components, easily track the relationships and interdependencies between managed elements and information throughout the network (i.e. "the Common Information Model (CIM)," page 2, section 2.1, second paragraph, lines 1-6).

Per claims 27-28, they are the system versions of claim 16 respectively, and are rejected for the same reasons set forth in connection with the rejection of claim 16 above.

Per claim 29:

Keller further teaches:

- a fault management application program (i.e. "root-cause," abstract; section 4.1, page 5, line 5).

Response to Arguments

8. Applicant's arguments with respect to claims 1-4, 6-10, and 16-35 have been considered but are moot in view of the new ground(s) of rejection.

9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Insun Kang whose telephone number is 571-272-3724. The examiner can normally be reached on M-F 8:30-5 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, MENG AI AN can be reached on 571-272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from

a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

IK
AU 2193